슈퍼히어로 기업 - 광고, 홍보, 전시 분야 기업

지자체, 국가기관, 대학교, 방송사, 기업 등 다양한 업체를 대상으로

TV CF/캠페인 영상 제작, 홍보/PR 분야 영상 컨텐츠 제작, SNS/Youtube 등 맞춤형 컨텐츠 제작

생성형 AI 응용 모델 종류

Text to Text - 텍스트를 입력 받아 텍스트 생성, 요약, 정리, 번역 등

Text to Image - 텍스트를 입력 받아 이미지로 변환하여 생성 및 편집

Text to Video(3D) - 텍스트 기반 비디오 또는 3D 형태로 변환하여 생성 및 편집

Text to Task - 작업 과정에서 문서를 변경 및 분석하거나 업무를 자동화하는 모델

머신러닝과 딥러닝의 차이

머신러닝 : 사용자가 설정한 명확한 특징을 기준으로 데이터를 입력해 해당 데이터들을 기준에 따라 분류, 예측하는 과정에서 패턴 학습, 비교적 적은 데이터와 계산량으로 학습 가능

딥러닝 : 머신러닝의 한 분야로 인공신경망을 활용하여 특징을 스스로 추출하고 학습, 대량의 데이터와 연산 과정이 필요하다는 단점 보유

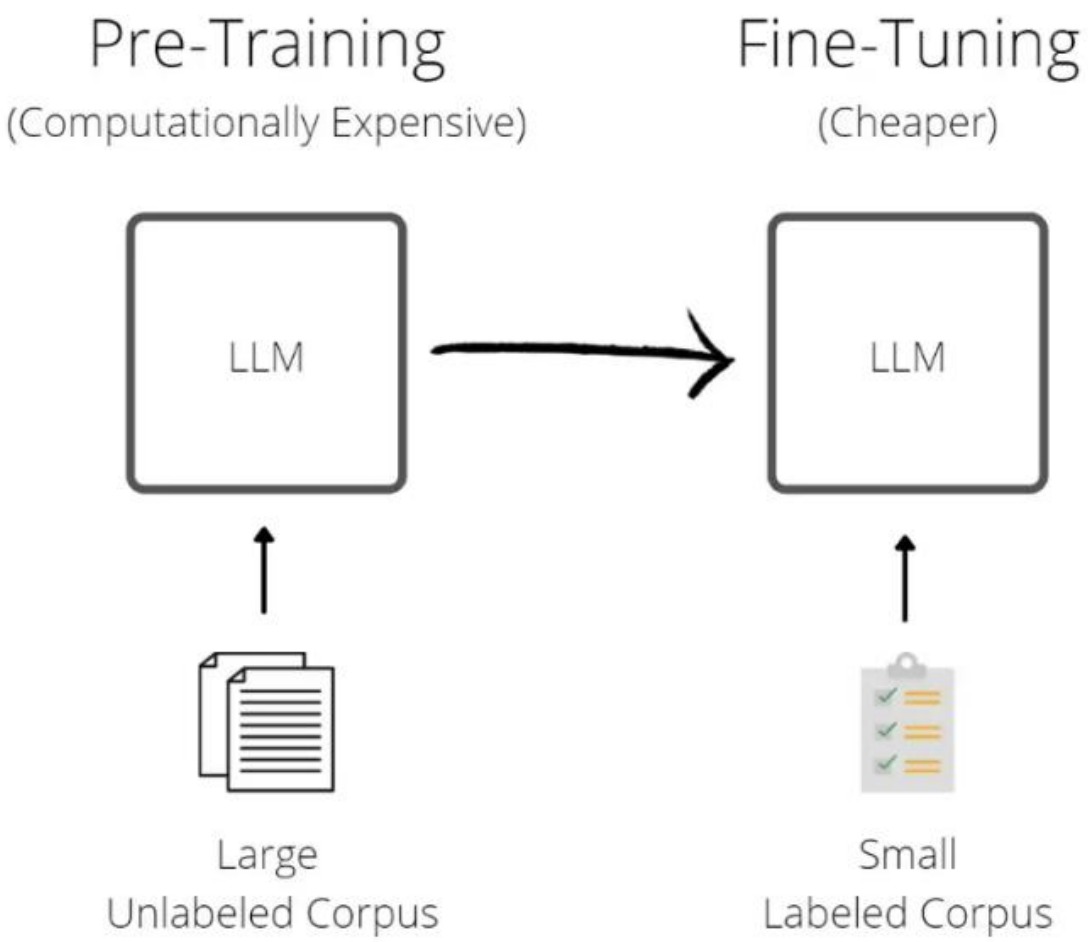
프롬프트 엔지니어링

생성형 AI 모델에게 의도한 답변을 이끌어 내기 위한 프롬프트를 설계, 조정하는 과정

AI에게 요청하는 사항에 따라 정확도, 창의성, 형식, 톤 등을 조절 가능

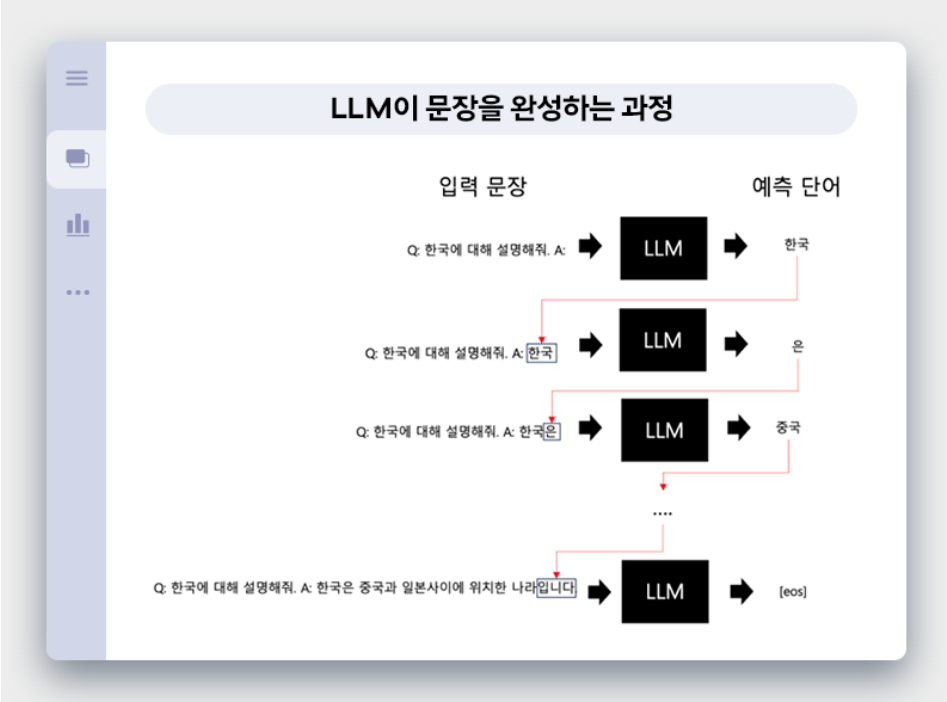
추가적인 학습 없이 응답 성능을 향상 시킬 수 있으나 기본적인 성능은 모델에 종속됨

사전학습과 미세조정(Fine Tuning)



방대한 양의 데이터 사전학습을 통해 문장 구조, 단어 의미, 문맥 등 언어 이해 능력을 습득하고 사전학습 모델을 다른 작업에 적합하도록 작은 데이터셋을 이용해 파라미터를 재조정시키는 미세조정을 통해 활용 가능

LLM 작동 원리



1. 데이터 입력
2. 토큰화(입력된 텍스트를 개별 단어, 문장 부호로 분리)

EX) "Hello, world!" → ["Hello", ",", "world", "!"]

3. 임베딩(토큰화된 단어를 임베딩 벡터로 변환 > 단어의 의미를 수학적으로 표시)

EX) "Hello" → [0.1, 0.3, 0.5, ...]

EX) "강아지"와 "개"는 비슷한 임베딩 벡터를 가지게 됨

4. 신경망 레이어(임베딩 벡터가 신경망 레이어를 통과하며 변환)
5. 어텐션 메커니즘(각 단어가 다른 단어와 상호작용하는 방식을 이해)
- 문장에서 어떤 단어에 집중해야 하는지 판단
6. 출력 생성(최종 신경망 레이어 출력을 기반으로 다음 단어, 문장 생성)

트랜스포머(인공신경망)

구글 딥마인드 연구진이 제안한 인공신경망 구조로 병렬 처리와 Attention 메커니즘을 통한 입력 데이터의 부분 강조를 통해 효율적인 자연어 처리 가능

1. Self-Attetion

➢ 문장의 각 단어가 다른 단어와 가지는 관계를 파악

EX) "나는 학교에 갔다" > "갔다"와 "학교"의 연관성 학습

2. Multi-Head Attetion

- 문장 안에 있는 모든 단어간의 관계 파악
- 쿼리(단어가 요구하는 정보), 키(다른 단어가 가진 정보), 밸류(중요도)

EX) "나는 밥을 먹었다" > 먹었다(쿼리/무엇을?), 밥(키), 밸류(연관성)

3. Positional Encoding

- 병렬 처리를 위해 문장을 순차적으로 입력 받지 않으므로 단어의 위치 정보를 따로 추가해 줄 필요가 있음

EX) "나는 밥을 먹었다" VS "밥을 먹었다 나는" 등 위치에 따른 언어의 의미나 문맥 파악

LLM 및 파인튜닝 활용 사례

1. 챗봇 형태

- 해당 분야의 전문 용어, 지식 등을 학습하여 답변 정확성 향상, 업무 효율성 향상으로 인한 인력 및 비용 절감

- LLM 모델에게 데이터를 학습시켜 맞춤형으로 업데이트한 파인튜닝, 특정 분야 데이터를 이용해 질문을 쿼리로 생성해 필요한 정보를 찾는 RAG 기반 검색 등 활용

EX) 애플케어 고객지원 센터(ASK) - 기술적 문제를 해결하기 위한 CS 특화 LLM 모델 'ASK'를 통해 고객지원 담당 직원의 답변 정확도 및 업무 효율 증가

EX) KB증권(스탁 GPT) - 챗GPT 기술과 주식 시장의 투자 정보를 활용하여 사용자에게 트렌드 파악, 종목 발굴, 이슈 검색 등 맞춤형 투자 조언 제공

2. 업무 자동화 형태

EX) 당근마켓 사례 - LLM을 이용한 서비스 질 향상 및 자동화

- 카테고리, 브랜드, 제품 등의 데이터를 학습시킨 LLM 모델을 통해 판매 글의 이미지와 게시글을 바탕으로 유사한 카테고리, 제품 추천

- 지도 데이터를 학습시킨 LLM 모델을 통해 게시글의 장소나 식당의 이름을 추출하고 지도에서 검색한 결과를 자동으로 표시

- 데이터 파이프라인을 활용해 각 LLM의 데이터들을 공유, 당근에서 실시간으로 생성되는 데이터 가공 및 활용 가능

3. 데이터 분석 AI

EX) LG전자(CHATDA)

- 제품의 운전 모듈을 분석하여 특정 제품이 특정 지역에서 어떻게 활용되고 있는 지 데이터 수집

EX) A 지역은 1 기능을 활용하는 빈도가 높다, B 지역은 2 기능을 덜 활용한다

- 해당 활용 데이터를 LLM 모델에게 학습시키고 분석하여 제품 기획 및 개발에 참고

현재 LLM 및 파인튜닝 기술의 문제점(단점)

1. 영미권에 맞춰진 LLM 기술

초기에 비해 한국어 성능이 향상되었으나, 아직은 프롬프트가 영어일 경우 더 좋은 성능을 보이는 LLM 모델

사용자 입장에선 한국어 프롬프팅이 편리하기 때문에 한국에서 생산된 데이터를 학습시켜 성능 차이를 개선하는 과정이 필요

2. 학습 데이터의 양과 품질에 의한 성능 의존

> 데이터 증강, 합성 데이터 등 기술 활용

3. 대규모 모델 파인튜닝을 위한 높은 GPU 및 메모리 비용

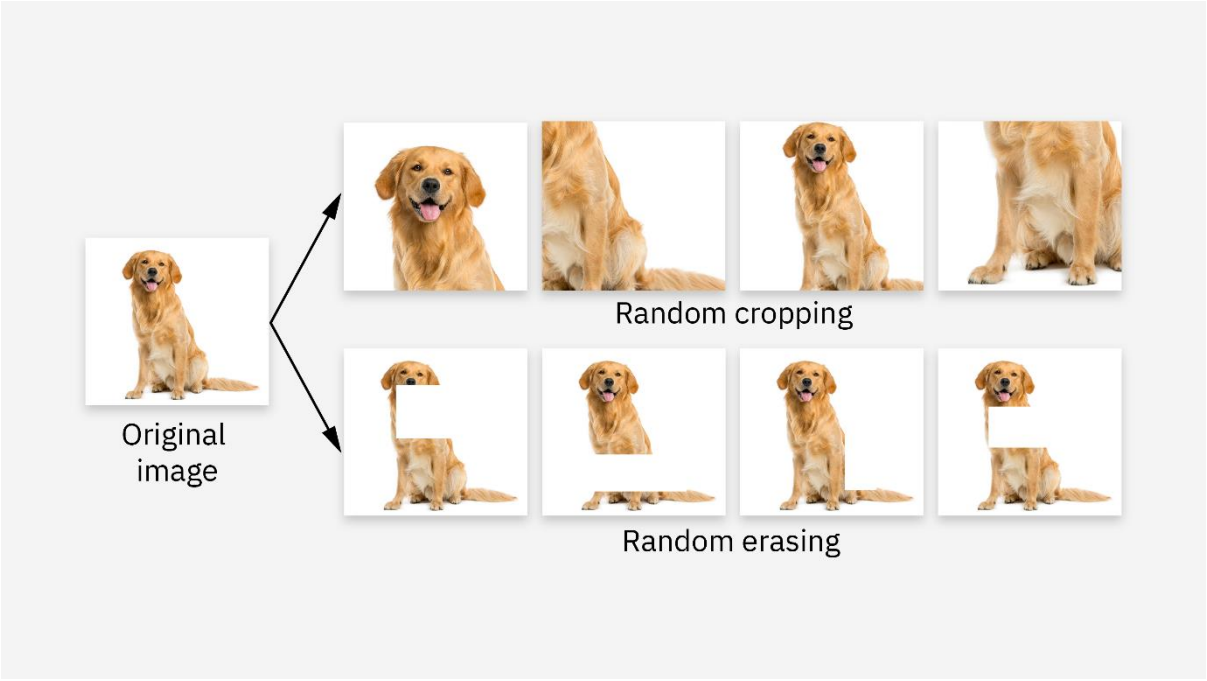
> LoRA

LLM 관련 기술

1) 전이학습 - 빅데이터로 미리 학습한 모델을 특정 작업에 맞게 추가 학습시킨 모델을 활용, 적은 데이터와 리소스로 성능 향상 가능
2) 지식 증류 - 큰 모델로 학습한 지식을 작은 모델에 전달하여 작은 모델의 성능을 향상, 적은 계산으로 높은 성능 유지 가능
3) 하이퍼 파라미터 튜닝 - 학습 과정에서 사용하는 파라미터(학습률, 배치 크기 등)의 최적화를 통해 모델 성능 극대화 (학습률, 배치 크기, 에포크(반복 수) 등)
4) LoRA(Low-Rank Adaptation) - 기존 모델의 파라미터를 고정하고 기존 모델의 일부 파라미터만 조정하여 적은 비용으로 파인튜닝, 파라미터를 줄여도 모델 성능

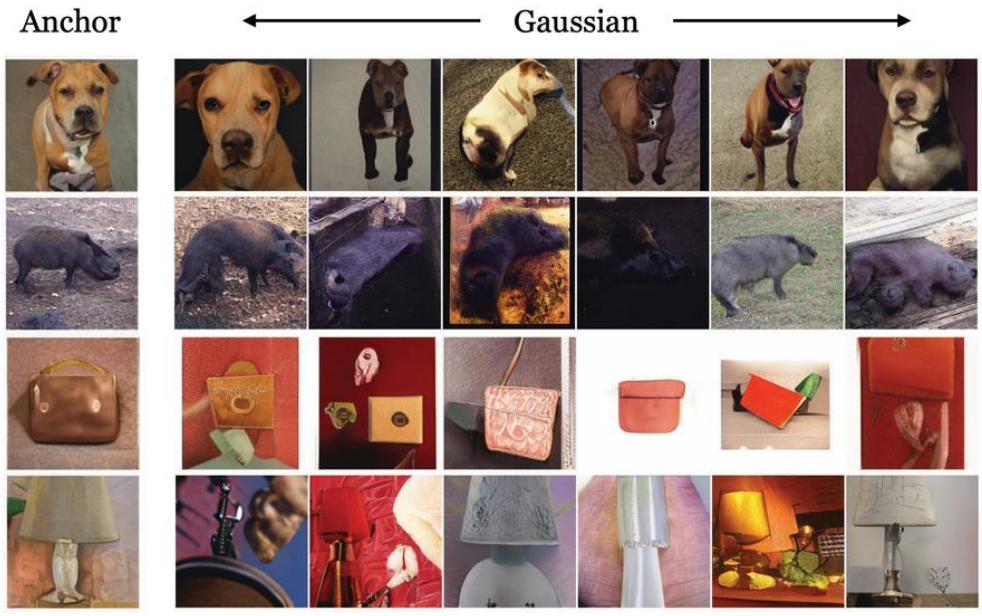
을 유지할 수 있어 학습 시간 단축 및 계산 자원 절약

5) 데이터 증강 - 원본 데이터를 변형하여 데이터를 생성해 추가하는 방법, 모델이 다양한 상황을 학습하여 일반화 성능을 향상



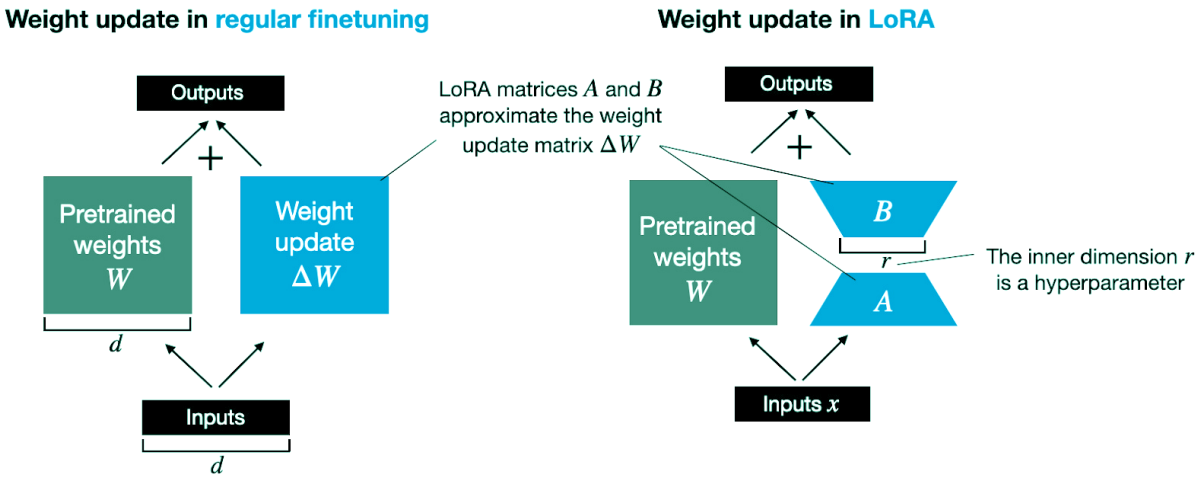
(데이터 증강)

6) 합성 데이터 활용 - 기존 LLM 모델을 이용해 가짜 데이터를 생성 후, 부족한 데이터 보충



(합성 데이터)

LoRA(Low-Rank Adaptation) : 기존 모델에서 일부 적은 수의 파라미터만 조정하여 효율적으로 파인튜닝하기 위한 인공지능 기법



LLM은 고성능일수록 많은 수의 파라미터를 가지고 있어 모든 파라미터를 파인튜닝 하는 것은 많은 계산과 시간을 요구하는 문제가 있다.(좌측의 경우)

따라서 조정할 일부 파라미터(우측의 A와 B)만 설정하여 학습

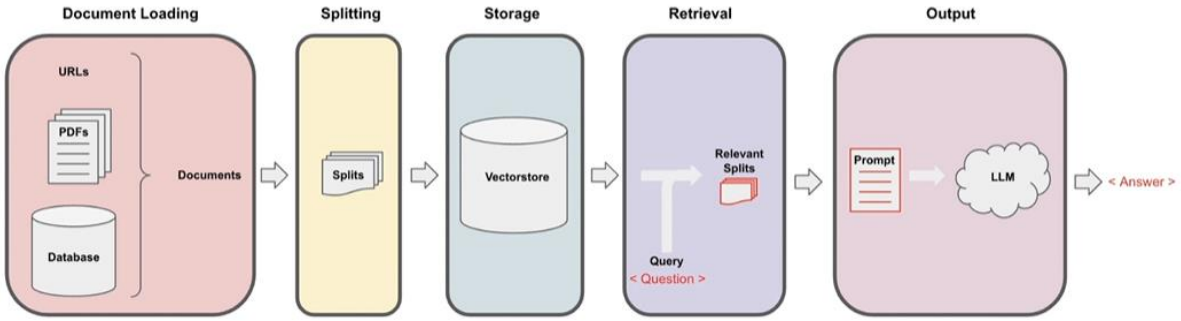
EX) 단어를 512 * 512로 표현되는 벡터로 변환한 LLM 모델을 기존의 파인튜닝으로 계산한다면 Input에 대해 512 * 512 개수의 파라미터를 학습해야 하나,

LoRA 방식을 이용하여 A와 B의 10(r)개 정도의 작은 파라미터만 학습시킨다면 학습하는 파라미터의 수는 2 * 10 * 512 (r 값은 선택)

단 비교하는 파라미터의 수가 줄어듬에 따라 r이 작다면 학습 결과의 디테일이 떨어지는 문제가 발생할 수 있음

RAG(검색 증강 생성) - 외부 데이터(웹, 논문 등)를 검색 후 응답 생성, 모든 정보를 사전 학습하지 않고 실시간으로 외부 데이터를 불러와 활용하기에 답변 최신화와 정확도 향상에 유리

[작동 방식]



0) 외부 데이터 텍스트를 추출하여 작은 단위로 분할
1) 분할한 문서 데이터를 벡터로 변환해 DB에 저장
2) 질문(쿼리) 입력
3) 질문을 벡터로 변환해 DB에서 유사한 문서 검색(임베딩)
   - 임베딩 : 단어, 문장을 숫자 형태로 변경
4) 질문에 관련된 외부 데이터를 탐색(문서 검색 알고리즘 활용)
5) 프롬프트와 외부 문서를 LLM에게 제공
6) GPT, BERT 등 생성 모델을 통해 프롬프트와 외부 데이터를 바탕으로 최종 응답 생성

결론적으로 기업 데이터는 파인튜닝으로 학습시키고

최근 트렌드나 자료들은 RAG 방식을 통해 답변하게 하는 방법이 괜찮아 보입니다.

또한, 파인튜닝을 위한 학습 데이터를 늘리고 효율적으로 학습시키기에 적합한 기법을 활용해보면 도움이 될 것 같아요.

유저 입력(프롬프트) > 프롬프트를 기반으로 외부 데이터 탐색(RAG) > 파인 튜닝된 LLM에게 프롬프트 및 외부 데이터 입력 > 좋은 응답